Attorney Docket No. 16869C-008600
Client Ref. No. HAL-ID 137

PATENT APPLICATION

Method and Apparatus for Encryption and Decryption in Remote Data Storage Systems

Inventors: **Kenji Yamagami**
Los Gatos, California 95032
Citizenship: Japan

Akira Yamamoto
Cupertino, California 95012
Citizenship: Japan

Naoko Iwami
Cupertino, California 95014
Citizenship: Japan

Masayuki Yamamoto
Sunnyvale, California 94087
Citizenship: Japan

Assignee: **Hitachi America, Ltd.**
Brisbane, California 94005
Incorporation: New York

Entity: **Large**

TOWNSEND and TOWNSEND and CREW LLP
Two Embarcadero Center, 8th Floor
San Francisco, California 94111-3834
(415) 576-0200

PATENT
Attorney Docket No. 16869P-008600
Client Ref. No. HAL-ID 137

Method and Apparatus for Encryption and Decryption in Remote Data Storage Systems

BACKGROUND OF THE INVENTION

This invention relates to information storage and retrieval, and in particular to encryption of data in storage systems having local and remote locations. In such systems, data are stored in a local storage system, for example, an array of hard disk drives, and data are also stored in a remote storage system. The use of a remote location for a copy of the data is desirable because it prevents loss of the data from corruption of communications links, natural disasters, or other causes. The remote copy function creates and maintains mirror volumes (duplicate sets) of the local data, but with the volumes of the sets separated by a "long" distance. The two disk systems are directly connected by remote links, through which updates to the data stored on the local disk system are copied to the remote disk system.

The remote system typically is coupled to the local system using communication links or a network, for example, ESCON, FC, TI, T3, ATM, etc. or a combination thereof, while suitable protocols are ESCON, SCSI, IP or others. In such a computing environment, data is exposed to the danger of corruption, theft and alteration because the network, or parts of the network, are publicly accessible, especially when using the Internet Protocol (IP).

Some companies, often referred to as storage service providers (SSP), provide a service to assist in managing customers' data. These companies sometimes rent their storage infrastructure and provide services such as storage management, remote copy, etc. to their customers. In such situations, the customers' data is stored in the SSP's storage system, and may be exposed to access by others.

U.S. Patents 5,459,857 and 5,544,347 describe remote copy technology which uses a remote link to connect two disk systems, enabling maintaining a duplicate copy, termed "a mirror," of the local system data on the remote disk system. The local disk system copies data on a local disk when duplication, termed "pair creation," is indicated. When a host updates data on the local disk, the local disk system transfers the

data to the remote disk system through the remote link. Thus no host operation is required to maintain a mirror of two volumes.

U.S. Patent 5,933,653 discloses a method for transferring data between a local disk system and a remote disk system. In a synchronous mode, the local disk system transfers data to the remote disk system before completing a write request from a host. In a semi-synchronous mode, the local disk system completes a write request and then transfers the write data to the remote disk system. Succeeding write requests are not processed until the previous data transfer is completed. With adaptive copy mode, data to be sent to the remote disk system is stored in a memory and transferred to the remote disk system when the local disk system and/or remote links are available for the copy task.

SUMMARY OF THE INVENTION

This invention provides a technique for assuring the privacy of a customer's data stored in a storage system. Encryption technology is employed in which a key for encryption and decryption is assigned to a volume or a set of volumes. Both the local and the remote disk system use the same key for a pair of volumes or a group of volumes. The keys are changeable without interrupting the host input/output operations to and from the local disk system. In addition, the keys can be periodically changed to improve security. The local disk system, which stores the initially created data, encrypts the data to be sent to the remote disk system and sends it to the remote disk system, where it is stored in encrypted form. To provide for selection of encryption and decryption, the local disk system and the remote disk system have a switching mechanism for implementing encryption and decryption. The disk systems can communicate with each other and change the encryption without losing the consistency of the remote copy.

In one embodiment of the invention, a method of controlling security of data in a storage system having a local disk system and a remote disk system includes performing certain steps in the local disk system and in the remote disk system. The steps performed in the local system include: when a write of data is to be made to the local disk system retrieving a previously stored encryption key, encrypting the data, and transferring the data to the remote disk system. The steps performed in the remote system include: retrieving the previously stored encryption key, determining an address for storage of the data, decrypting the data, writing the decrypted data in the remote disk system; and notifying the local disk system that the step of writing the decrypted data is complete.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is a block diagram illustrating the overall configuration of a system according to a preferred embodiment of this invention;

Figure 2 is an exemplary encryption control table;

Figure 3 is a flowchart illustrating the encryption and decryption process;

Figure 4 is a flow chart illustrating a first method of transparent key exchange;

Figure 4b illustrates the concept behind transparent key exchange;

Figure 5 is a flow chart illustrating a second method of transparent key exchange;

Figure 6 is a flow chart illustrating a first method of controlling encryption;

Figure 7 is a flow chart illustrating a second method of controlling encryption; and

Figure 8 is a flow chart illustrating a third method of transparent key exchange.

DESCRIPTION OF THE SPECIFIC EMBODIMENTS

In a system according to an embodiment of this invention, encryption is enabled for a storage system having both local and remote disk systems. The assignment of encryption keys to volumes is first discussed with respect to Figure 1. Two disk systems, referred to as the local disk system 100 and the remote disk system 110, each include one or more hard disk drives 102, 112, optical storage disks, flash memories, or other storage media. While the following description refers to disks, it should be understood that any type of data storage media can be employed. Each disk system also has processors (not shown) on which appropriate software programs run, additional memories for storing control data and tables for the software, etc. One or more host computers 115 connect to at least the local disk system 100, by the connection of SCSI 122, Fibre, ESCON, etc. The host computer 115 accesses the disks in the local disk system through the connection 122. One or more host computers 118 also may be connected to the remote disk system 110.

Management consoles 125, 130 provide connections to the local, and optionally to the remote disk system, using LAN 133, proprietary connection 135, SCSI, Fibre or ESCON, or other well known technique. An administrator manages the disk systems through this management consoles 125, 130. If desired, the management console 125 for the local disk system also may connect to the remote disk system. The connection between the local and remote disk systems may comprise ESCON, SCSI, LAN/WAN or Fibre 140, or combination of them, for example, using a gateway appliance. As shown in Figure1, a key is assigned to a volume or a group of volumes. The same key is assigned to a local volume (or a group of local volumes) and to a remote volume (or a group of remote volumes). One can arbitrarily define groups of volumes. For example, one may define a group of volumes deploying an entire database.

The local 100 and remote 110 disk systems maintain an encryption control table 200 as depicted in Figure 2. Each entry in the table is indexed by a volume number 240, thus allowing a separate key to be assigned to each volume. If a key is assigned to a group, entries indexed by volume number of the group will have the same value for the key 210. The value of key 210 for a volume is the same in both the local disk system and remote disk system. The column designated key 210 shows the key assigned to the volume listed in the column labeled volume 240, while the encryption 220 and decryption 230 columns indicate the status of encryption, as follows. A "Yes" in column 220 indicates the local system encrypts the data before sending it to the remote disk system. A "No" in column 220 indicates the local system sends ordinary (non-encrypted data) to the remote disk system. With respect to column 230, a "Yes" in column 230 indicates that the remote system must decrypt the data before using it, while a "No" in column 230 indicates that the remote copy data has been stored in decrypted form and therefore can be used without decryption.

Figure 3 is a flowchart of the encryption and decryption process. Three situations will invoke the remote copy process depicted in Figure 3. First, when establishing a pair (referred to herein as initial copy), the local disk system 100 copies all data on the local disk to the remote disk 110. An administrative controller usually provides the local and remote disk addresses, and both local and remote disk systems store this information. Second, when a host updates data to be stored in a local disk 100, the local disk system transfers the new (changed) data to the local disk, then the local disk system transfers the changed data to the remote disk system. The host provides the

location of the data in the form of the local disk address. Third, when the local disk system schedules copying data to the remote disk, the local disk system transfers the data to copy, together with the location of the data and the disk address.

The desired remote disk address can be retrieved from the local disk system. As described previously, the local disk system has stored the relationship between the local disk or volume and the remote disk or volume when the administrator established a pair. This enables the remote disk address to be located. By referring to the appropriate entry in the encryption control table corresponding to the address, the remote disk system locates the key for the disk. The local disk system knows its local disk address. By referring the entry corresponding to the address in the encryption control table 200, it finds the correct key for the disk. Steps 310-330 illustrate locating the right key at the remote disk system. A write request from the local disk system to the remote disk system includes the remote disk address. Once the address is located, the data is sent to the remote disk, decrypted, and stored, all as shown by steps 330-340. When the write at the remote disk is complete, a message 350 is sent to the local disk system, informing it of the completion.

There are two methods enabling keys to be changed without interrupting host operations. Because the remote system will be operating at least slightly later than the local system, there will be time differences in the writing of data at the two locations. This makes it undesirable to just change the key at a designated time. If this were to occur, the key exchange might be performed in the middle of an operation.

Changing enabling keys without interrupting host operations is referred to herein as "transparent key exchange." In the first implementation, illustrated by Figure 4, the local disk system counts the number of I/O requests from the local disk system to the remote disk system for each volume pair. (See step 430.) When an administrator introduces a new key and initiates key exchange through the management console, the local and remote disk systems perform the operations shown in the flowchart in Figure 4. In particular, a boundary number is determined which corresponds to the I/O number after which the key is to change. Upon detection of this number of I/O operations in the local disk system, the key is changed. Similarly, upon detection of this number of I/O operations in the remote disk system, the key is also changed.

Figure 4b illustrates this process conceptually. The upper time line illustrates operations in the local system, while the lower time line illustrates

corresponding operations in the remote system, and that those operations lag the operations in the local system. Note that the key is changed after operation 4 in each of the local and the remote system, and that this change in key occurs at a different time in each system. As illustrated in Figure 4, the request and/or data, sent from local to remote at steps 410 and 440, are encrypted and decrypted by the current key, not the new key.

The copy process is running during the operations in Figure 4. Therefore I/O requests from local to remote are being processed in parallel with the key change operation. The local disk system must choose an appropriate I/O number at step 440. It then prevents performing the I/O with that number until step 440 completes.

A second method of implementing key exchange, illustrated in Figure 5, is by using a pair control mechanism such as splitting and re-synchronizing mirrored pairs. When splitting a mirror, the local disk system stops copying data to the remote disk system. The local disk system maintains a list of updates from hosts to the local volume, usually by using a pending bit map. When re-synchronizing the mirror, the local disk system begins copying pending data to the remote volume by referring to the bit map.

In the embodiment in which key exchange is performed using the process of splitting and re-synchronizing a mirrored pair, an administrator provides the new key and instructs key exchange through the management console. Then the local and remote disk systems perform the operations in Figure 5. At step 530 the local disk system changes its pair status, stops copying data to the remote disk system and begins marking the bit map. The pair status for both local and remote volumes changes to "Suspend," which means data between local and remote disks is not equivalent. In some implementation, this process may cause the local disk system to communicate with the remote disk system (step 540). At step 550, to validate the new key, the local and the remote disk system store the new key in the encryption table 210. Then at step 570, after re-synchronizing the pair, the local disk system changes its pair status and restarts copying in accordance with the bitmap. When the host updates data, the data is also copied to the remote system. The pair status switches to "Copy Pending," which means copy in progress, and then to "Pair," meaning that the data between local and remote disks is equivalent. In some implementations, this process may cause the local disk system to communicate with the remote disk system (step 580). The remote disk system also changes the pair status to "Copy Pending" and then "Pair."

The use of encryption or decryption is controllable. Encrypting data may cause performance degradation, and some data does not need encryption. The choice of whether to encrypt or not is a tradeoff between importance of data and performance, and is left to the users' decision. This invention enables the user to choose whether to use encryption and/or decryption. There are two methods enabling turning encryption and decryption on and off. These techniques are depicted in Figure 6 and 7. They use the encryption table of Figure 2.

If a user selects "encryption=YES and decryption=NO" (meaning that the remote data is stored encrypted) the methods for changing a key described in Figure 4 and 5 need to be modified. Before changing the key, the data stored in the remote disk was encrypted by a first key. When the key is changed, the data is encrypted by a second key and stored in the remote disk. This implies data encrypted by two or more different keys are present on the remote disk. Although feasible, it is generally undesirable to maintain different keys for each encrypted portion of the remote disk. To solve this problem, the remote disk system re-encrypts all data on the remote disk with the new key. A predetermined amount of data, e.g. a track, is read from the disk to the cache memory of the remote disk system, decrypted by the current key, encrypted by the new key, and then stored back to the same location on the remote disk. The remote disk system keeps track of this process with a bit map. If the local disk system copies data to a location that has not finished re-encryption, the remote disk system performs the above operation before responding to the local disk system.

Figure 8 illustrates the above process in detail. As shown, after initializing the bitmap at step 800, a copy request (a write I/O request) 810 indicates the location of records or blocks to be updated. For example, with the CKD protocol, the location is a track address and a record number of the heading record, together with the number of records, while with the SCSI protocol, a block address of the heading block and number of blocks are provided. At step 890, the remote disk system does step 840 to 860 for the track(s) that contain the records or the blocks. At step 870 the re-encrypted data is written to the disk.

The apparatus and methods described in this invention encrypt and decrypt data being transferred between two disk systems. A key for encryption and decryption is assigned to a volume. This protects remote copy data from being misappropriated and/or altered. An administrator can manage encryption because the remote copy is done for a

pair or a group of pairs of volumes. This invention also provides a method to change keys transparently. In addition, the invention provides a method that enables an administrator to choose when to use encryption and/or decryption, even on a volume by volume-pair basis.

The preceding has been a description of the preferred embodiment of the invention. It will be appreciated that deviations and modifications can be made without departing from the scope of the invention, which is defined by the appended claims.